Lab Assignments Computational Finance

Lecturer: Prof. dr. Drona Kandhai

Teaching assistant 1: Alex Janczewski
E-mail: a.t.janczewski@uva.nl

Teaching assistant 2: Sven Poelmann
E-mail: s.p.poelmann@uva.nl

Submission guidelines
These assignments can be done in groups of three students. Reports with a *clear description of the assignment, the methods, the results and discussion* should be submitted before the deadlines. Quality is valued over quantity: aim for a report with a page count of 5-10. You are free to choose the programming language/environment in which you would like to write your computer programs. If you have questions about the assignments do not hesitate to contact the teaching assistants.

Grading scheme

- Each of the three assignments carries equal weight of 20% and the exam is worth 40%;
- The score of the exam should be 5 points (on the scale of 1 to 10) and higher for passing the course;

Assignment 1	Assignment 2	Assignment 3	Exam
20%	20%	20%	40%

Assignment 1: Black-Scholes and the binomial tree

Part I
Theory: derivatives and no-arbitrage

A common assumption for financial models, is that money invested in the money-market or bank-account yields interest at a constant, risk-free rate r. The frequency at which interest is compounded and added into the balance, is defined as the *compounding type*. Assume that someone invests €100 in a bank-account at a risk-free rate of 6%. Below are a few examples of what his balance B would be after one year, for different compounding types:

- **Annual compounding:** $B = 100 \times (1 + 0.06) = 106$
- **Semi-annual compounding:** $B = 100 \times \left(1 + 0.06 \times \frac{1}{2}\right)^2 = 106.09$
- **Monthly compounding:** $B = 100 \times \left(1 + 0.06 \times \frac{1}{12}\right)^{12} = 106.17$

1. In theory, continuous compounding means that an account will be compounding interest over an infinite number of periods per year, which is continuously added into the balance. Prove that an amount C invested in the money-market at continuous compounding has a value of $C \times e^{r\Delta t}$ after a period Δt.

A *coupon bond* is a debt security under which the issuer owes a dept to the holder. The issuer is obligated to pay periodic interest and must return the principal amount at maturity. In other words, it is a financial product where the holder receives a sequence of constant payments (so-called coupons) at fixed intervals from the issuer until the bond matures and the principal is repaid. The coupon rate is not necessarily equal to the risk-free rate of the money-market.

2. Consider a coupon bond with a principal of €50.000, a maturity of 2 years and quarterly (i.e. every 3 months) coupons of €300. Assume that the money-market has a risk-free rate of 1.5% at continuous compounding. Calculate the fair-value of this coupon bond.

A *forward* is a contract between two parties to buy or sell an asset S at a specified future date T for a specified future delivery price K. The party that assumes the *long position* in the forward has the obligation to buy the asset from the party that assumes the *short position.* At the entry of the contract ($t = 0$), the delivery price K is specified to be equal to the *forward price* F, which means the forward has zero initial value.

3. Use the no-arbitrage principle to argue that the forward price of a contract at time zero is equal to $F_0 = S_0 e^{rT}$.

Let C_t and P_t denote the value of a European call and put option respectively, written on a stock S with similar strike price K and maturity T. Consider two portfolios, which at time zero contain: 1) A call option and an investment of Ke^{-rT} in the money-market; 2) A put option and one share of the stock S.

4. Draw the pay-off diagrams for both portfolios, showing the profit at maturity as a function of S_T. Explain the figures.

5. Use the no-arbitrage principle to argue that for $t \in [0, T]$ the following relation, known as *the put-call parity*, must hold:

$$C_t + e^{-r(T-t)}K = P_t + S_t$$

Part II
The binomial tree: option valuation

A commonly used approach to compute the price of an option is the so-called binomial tree method. Suppose that the maturity of an option on a non-dividend-paying stock S_t is divided into N subintervals of length Δt. We will refer to the j^{th} node at time $i \cdot \Delta t$ as the (i, j) node. The stock price at the (i, j) node is $S_{i,j} = S_0 u^j d^{i-j}$ (with u and d the upward and downward stock price movements, respectively). In the binomial tree approach, option prices are computed through a backward induction scheme:

1. The value of a call option at its expiration date is $\max\{0, S_{N,j} - K\}$;
2. Suppose that the values of the option at time $(i+1) \cdot \Delta t$ is known for all j. There is a risk-neutral probability p of moving from the (i, j) node at time $i \cdot \Delta t$ to the $(i+1, j+1)$ node at time $(i+1) \cdot \Delta t$, and a probability $1 - p$ of moving from the (i, j) node at time $i \cdot \Delta t$ to the $(i+1, j)$ node at time $(i+1) \cdot \Delta t$. Risk-neutral valuation gives

$$f_{i,j} = e^{-r\Delta t}\left(p f_{i+1,j+1} + (1-p) f_{i+1,j}\right)$$

A general introduction to the binomial tree model and a detailed derivation of the relevant parameters is provided in appendix A. *Please read the appendix carefully before starting on the assignment.*

Consider a European call option on a non-dividend-paying stock with a maturity of one year and strike price of €99. Let the one year interest rate be 6% and the current price of the stock be €100. Furthermore, assume that the volatility is 20%.

1. Write a binomial tree program to approximate the price of the option. *To help you get started, you can use the instructions, templates and hints provided in appendix B.* Construct a tree with 50 steps and explicitly state your option price approximation in the report.
2. Investigate how your binomial tree estimate compares to the analytical Black-Scholes value of the option. Do experiments for different values of the volatility. *The Black-Scholes formula for European option prices is treated in appendix C: please read that carefully.*
3. Study the convergence of the method for increasing number of steps in the tree. What is the computational complexity of this algorithm as a function of the number of steps in the tree?

4. Compute the hedge parameter Δ from the binomial tree model at $t = 0$. Compare with the analytical Black-Scholes delta $\Delta_0 = N(d_1)$. Experiment for different values of the volatility.

5. Now suppose that the option is American. Change the code such that it can handle early exercise opportunities. What is the value of the American put and call for the corresponding parameters? Experiment for different values of the volatility.

Part III

Black-Scholes model: hedging simulations

The fundamental idea behind the Black-Scholes model is that of dynamic replication of the claim by taking positions in the underlying. In practice this means that a trader should apply a dynamic hedging strategy in order to ensure that the claim is replicated at expiry. In this part of the assignment we will apply a delta hedging of a European Call option.
We make the following assumptions:

- The dynamics of the stock price S is given by the following equation

$$dS_t = rS_t dt + \sigma S_t dz_t$$

- The option and the corresponding delta sensitivities is based on the Black-Scholes model.

A general introduction to the Black-Scholes model, a derivation of the option pricing formula and instructions on simulation techniques are provided in appendix C. *Please, read the appendix carefully before starting on the assignment.*

1. Let C_t denote the risk-neutral price of a European call option in the Black-Scholes model. The delta-parameter is defined as $\Delta_t := \frac{\partial C_t}{\partial S_t}$. Show that in the Black-Scholes model $\Delta_t = N(d_1)$.

2. Let P_t denote the risk-neutral price of a European put option. Use the put-call parity and the call option price given in appendix C, to show that P_t is given by

$$P_t = e^{-r\tau} K N(-d_2) - S_t N(-d_1)$$

Consider again a short position in a European call option on a non-dividend-paying stock with a maturity of one year and strike price K of €99. Let the one year interest rate be 6% and the current price of the stock be €100. Furthermore, assume that the volatility is 20%.

3. Use the Euler method to perform a hedging simulation. Do an experiment where the volatility in the stock price process is matching the volatility used in the delta computation (set both equal to 20%). Vary the frequency of the hedge adjustment (from daily to weekly) and explain the results. Perform numerical experiments where the volatility in the stock price process is not matching the volatility used in the delta valuation. Experiment for various levels and explain the results.

Appendices: Theory

In the sections below, some additional theoretical background for the assignment is presented. General introductions to the binomial tree model and the Black-Scholes model are treated. Additionally, we provide instructions and hints to get you started with the coding of the first assignment. Please read these appendices carefully before starting on the assignment.

A The binomial tree: model theory

The binomial tree is a discrete-time model for the evolution of asset prices. It is an elegant tool to calculate option values and hedge parameters. Here we will discuss the assumptions to the model, provide some intuition to it and derive the main model parameters.

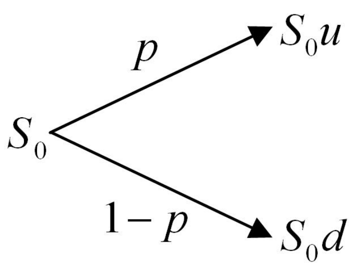


Figure 1: Stock price movements

The binomial tree is a simplified representation of the market. Let S denote the value of a stock. The main concept is that after each time period Δt, the price of the stock can make two movements, as depicted in Figure 1. We refer to these as the *up movement*, in which case $S_{t+\Delta t} = S_t \times u$ or the *down movement*, in which case $S_{t+\Delta t} = S_t \times d$, for two positive numbers $0 < d < u$. Obviously, stock price movements are in reality processes of a much higher complexity. Still, this simple model is useful in practice because it is computationally tractable and a relatively good approximation to more realistic continuous-time models if a sufficient number of periods is used.
Below we summarize the main assumptions of this model:

1. Cash invested in the money-market yields a continuously compounded interest at a constant rate r.
2. A stock price S_1 after a period of time Δt can only have two possible outcomes: $S_0 \cdot u$ or $S_0 \cdot d$ with $0 < d < e^{r\Delta t} < u$.
3. The economy is free of arbitrage.
4. There are no transaction costs for trading shares of stock.

We will use this model to introduce the concept of a risk-neutral price and a replication portfolio.

A.1 Example: Risk-neutral option pricing with a one-period tree

We start with a simple one-period example, taken from [1]. Consider a stock of which the value is denoted S_t. We assume a highly simplified market, captured by a one-period binomial tree. Suppose $S_0 = 4$, $u = 2$, $d = 0.5$ and assume that the accrued interest over Δt is given by

$e^{r\Delta t} = 1.25$. Therefore, the two possible price moves of the stock over Δt are $S_0 \times d = 2$ and $S_0 \times u = 8$ (see Figure 2).

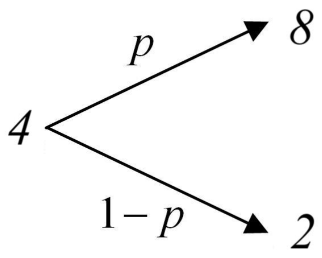


Figure 2: Example of stock price movements

Secondly we consider a European call option written on one share of stock S_t. A European option is a contract which gives the holder the right, but not the obligation, to buy the underlying asset at a specified strike price on a specified future date. Suppose that the expiration of the option is after one period Δt and the strike price is $K = 5$. The central question is now: **What is the fair value f_0 of this option contract at time zero, before the price movement of the stock is known?**
We argue that the answer to this question is $f_0 = 1.20$, using a *no-arbitrage* argument. Suppose that an agent just sold the option contract for the price of 1.20. At $t = 0$ he buys $\Delta = 0.5$ of the stock. That means the agent has a portfolio with 0.5 shares of stock and a cash-position of $f_0 - \Delta \times S_0 = 1.20 - 0.5 \times 4 = -0.80$ (i.e. a debt). Now, at time $t = 1$ due to the accrual of interest, the agent has a debt of $0.8 \times e^{r\Delta t} = 0.8 \times 1.25 = 1$. For the stock value two scenarios can occur:

- **Up movement:** The buyer of the option exercises the contract, pays the strike price to the agent and receives one unit of stock in return. From the agent's perspective this means he has to buy 0.5 shares (in addition to the 0.5 shares he already had in his portfolio) at the market-price of that time. That costs him $(1 - \Delta) \times S_1 = 0.5 \times 8 = 4$ and creates a total debt of $1 + 4 = 5$. This then exactly cancels against the received strike price of $K = 5$. Hence, the agent is left with a position of zero.

- **Down movement:** The buyer of the option does not exercise the contract, because it would yield him a loss. The agent therefore sells his 0.5 shares in the market. That earns him $\Delta \times S_1 = 0.5 \times 2 = 1$, which exactly cancels against his debt of 1. Hence, the agent is again left with a position of zero.

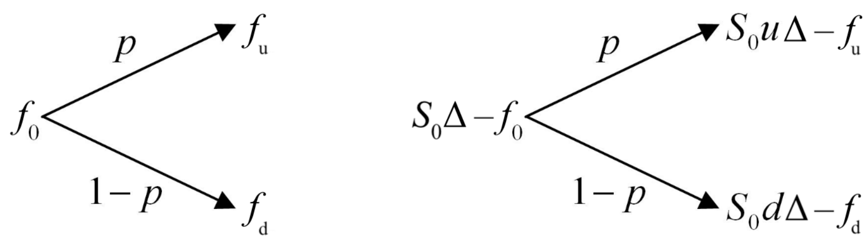
The conclusion is that an agent can set up a portfolio of assets and cash that exactly replicates the value of the option. This is called *hedging*. Should the option be sold at a higher or lower price, it means that someone can make a guaranteed profit, which we refer to as *arbitrage*. $f_0 = 1.20$ is therefore the *no-arbitrage price* or the *risk-neutral value* of the option.

A.2 Derivation binomial tree parameters

We will step by step derive the parameter values that are relevant to the binomial tree model.

A.2.1 The delta-parameter (Δ)

We start with Δ, which represents the amount of shares that need to be bought at $t = 0$ by the agent in order to set up a hedge. Figure 3a depicts the price movements of the risk-neutral



(a) The option value (b) The portfolio value

Figure 3: Price movements in the one-period binomial tree

option value. Figure 3b depicts value movements of the agent's portfolio. At time zero, the agent sells the option contract for the price f_0 and buys Δ shares of the stock at price S_0. After time Δt, the portfolio is worth $S_0u\Delta - f_u$ in case of an up movement and $S_0d\Delta - f_d$ in case of a down movement.

The key insight is that the agent wants to choose Δ such that the final value of his portfolio is independent of the price movement of the stock. The portfolio is then *risk-free.* This is the case if and only if

$$S_0u\Delta - f_u = S_0d\Delta - f_d \tag{1}$$

Solving for Δ yields

$$\boxed{\Delta = \frac{f_u - f_d}{S_0u - S_0d}} \tag{2}$$

A.2.2 The risk-neutral probability (p) and fair value (f_0)

The fair value f_0 can be determined by noting that the agent should have a zero position at maturity of the option as otherwise the agent must have obtained a loss or gain. That would imply an arbitrage since the portfolio is set up to be risk-free. At time zero, the cash-position of the agent is $f_0 - \Delta S_0$. At maturity this cash has accrued interest, by which it has the value $e^{r\Delta t}(f_0 - \Delta S_0)$. His portfolio at maturity is worth $S_0u\Delta - f_u$ in case of an up movement and $S_0d\Delta - f_d$ in case of a down movement, which by the deliberate choice of Δ will be equal. The final position of the agents is zero if his cash-position and portfolio value add up to zero at maturity. Therefore it follows that

$$\begin{aligned} & e^{r\Delta t}(f_0 - \Delta S_0) + (S_0u\Delta - f_u) = 0 \\ \implies & f_0 = S_0\Delta - e^{-r\Delta t}(S_0u\Delta - f_u) \end{aligned}$$

Having derived f_0, we aim to rewrite it in a convenient form by substituting the delta-parameter. Recall that $\Delta = \frac{f_u - f_d}{S_0 u - S_0 d}$. If we substitute that in the expression for f_0, we find:

$$\begin{aligned}
f_0 &= S_0\Delta - e^{-r\Delta t}\left(S_0 u\Delta - f_u\right) \\
&= e^{-r\Delta t}\left(S_0\Delta\left(e^{r\Delta t} - u\right) - f_u\right) \\
&= e^{-r\Delta t}\left(S_0\frac{f_u - f_d}{S_0 u - S_0 d}\left(e^{r\Delta t} - u\right) - f_u\right) \\
&= e^{-r\Delta t}\left(\left(\frac{e^{r\Delta t} - u}{u - d} - 1\right)f_u - \frac{e^{r\Delta t} - u}{u - d}f_d\right) \\
&= e^{-r\Delta t}\left(\frac{e^{r\Delta t} - d}{u - d}f_u + \frac{u - e^{r\Delta t}}{u - d}f_d\right)
\end{aligned}$$

The parameter p is defined as

$$\boxed{p := \frac{e^{r\Delta t} - d}{u - d}} \tag{3}$$

Note that the following relation holds

$$1 - p = \frac{u - d}{u - d} - \frac{e^{r\Delta t} - d}{u - d} = \frac{u - e^{r\Delta t}}{u - d}$$

Therefore we can rewrite the expression for f_0 in the following convenient form

$$\boxed{f_0 = e^{-r\Delta t}\left(pf_u + (1-p)f_d\right)} \tag{4}$$

We refer to the quantities p and $1 - p$ as the *risk-neutral probabilities* of an up and down movement respectively. By interpreting p as a probability, the fair value of the option can be interpreted as the *expected value* of the option's pay-off. In other words, we can write:

$$f_0 = \mathbb{E}\left[e^{-r\Delta t}f_1\right] \tag{5}$$

Also note that as a consequence, the risk-neutral expectation of S_1 can shown to be equal to

$$\begin{aligned}
\mathbb{E}\left[S_1\right] &= pS_0u + (1-p)S_0d \qquad (6)\\
&= \frac{e^{r\Delta t} - d}{u - d}S_0u + \frac{u - e^{r\Delta t}}{u - d}S_0d \qquad (7)\\
&= S_0e^{r\Delta t} \qquad (8)
\end{aligned}$$

A.2.3 The up and down factors (u and d)

We finalise by deriving the u and d factors. These factors should be chosen such that they represent market conform price movements. To do so, the factors are linked to the volatility of the stock price. The volatility σ is a measure for the degree of variation in the stock price over time. The derivation in this section is based on the following two assumptions:

- For small values of Δt, the variance of the stock price change is approximately $S_0^2\sigma^2\Delta t$.
- $u \times d = 1$.

Recall that $\mathbb{E}[S_1] = S_0 e^{r\Delta t}$, then according to the definition of the variance we have

$$\begin{aligned}\operatorname{Var}(S_1) &= \mathbb{E}\left[S_1^2\right] - \mathbb{E}[S_1]^2 \\ &= p\,(S_0 u)^2 + (1-p)\,(S_0 d)^2 - \left(S_0 e^{r\Delta t}\right)^2\end{aligned}$$

Now substitute our derived value of p to find

$$\begin{aligned}\operatorname{Var}(S_1) &= S_0^2\left(\frac{e^{r\Delta t}-d}{u-d}u^2 + \frac{u-e^{r\Delta t}}{u-d}d^2 - e^{2r\Delta t}\right) \\ &= S_0^2\left(e^{r\Delta t}(u+d) - ud - e^{2r\Delta t}\right)\end{aligned}$$

The first assumption says $\operatorname{Var}(S_1) = S_0^2\sigma^2\Delta t$. Divide both sides by S_0^2 and substitute $d = \frac{1}{u}$, which follows from the second assumption. We can then write

$$\begin{aligned}\sigma^2\Delta t &= e^{r\Delta t}\left(u + \frac{1}{u}\right) - 1 - e^{2r\Delta t} \\ u + \frac{1}{u} &= e^{-r\Delta t}\left(\sigma^2\Delta t + 1 + e^{2r\Delta t}\right) \\ &= e^{-r\Delta t}\sigma^2\Delta t + e^{-r\Delta t} + e^{r\Delta t}\end{aligned}$$

Now use Taylor expansions to approximate $e^{-r\Delta t} \approx 1 - r\Delta t$ and $e^{r\Delta t} \approx 1 + r\Delta t$. If we neglect all terms with $(\Delta t)^2$ and higher powers, we have

$$\begin{aligned}u + \frac{1}{u} &\approx (1 - r\Delta t)\,\sigma^2\Delta t + (1 - r\Delta t) + (1 + r\Delta t) \\ &\approx \sigma^2\Delta t + 2\end{aligned}$$

This we can be rewritten as

$$u^2 - \left(\sigma^2\Delta t + 2\right)u + 1 = 0$$

which can be solved by using the quadratic formula and again ignoring higher powers of Δt:

$$\begin{aligned}u &\approx \frac{\sigma^2\Delta t + 2 \pm \sqrt{\left(\sigma^2\Delta t + 2\right)^2 - 4}}{2} \\ &\approx \frac{1}{2}\sigma^2\Delta t + 1 \pm \sigma\sqrt{\Delta t}\end{aligned}$$

The solution with the minus-sign corresponds to the d parameter $\left(= \frac{1}{u}\right)$, therefore we will discard it for now. As a final step, consider the second-order Taylor expansion of $f(x) = e^{\sigma x}$ around zero. This would yield $f(x) \approx 1 + \sigma x + \frac{1}{2}\sigma^2 x^2$. Therefore it follows that

$$\begin{aligned}u &\approx 1 + \sigma\sqrt{\Delta t} + \frac{1}{2}\sigma^2\Delta t \\ &\approx f\left(\sqrt{\Delta t}\right) = e^{\sigma\sqrt{\Delta t}}\end{aligned}$$

Hence we conclude:

$$\boxed{u = e^{\sigma\sqrt{\Delta t}} \qquad \text{and} \qquad d = e^{-\sigma\sqrt{\Delta t}}} \tag{9}$$

B The binomial tree: coding instructions

In this section, we do not discuss the theory behind the binomial tree, but provide you with a basic setup to implement your binomial tree in Python. Note that this is a setup, the code as presented here is not yet working. In order to calculate the answers, missing parts need to be completed.

B.1 Running time

You might be tempted to implement this as a tree recursion algorithm, but unfortunately this leads to sub-optimal running times $O(2^N)$. And you will quickly discover that running for larger N becomes intractable.
Therefore, we suggest a different approach. We will calculate the numbers in the binomial tree, but store them as entries of a matrix.
For example, a tree with $N = 2$ looks as follows ($S_0 = 80, \sigma = 0.1, T = 1, N = 2$):

```
              92.15
            /
       85.86
     /      \
80            80
     \      /
       74.53
             \
              69.44
```

And we can represent the matrix with the values as:

```
    [ 80.      0        0    ]
    [ 74.53    85.86.   0    ]
    [ 69.44.   80.      92.15]
```

Thus, the matrix contains all the required values, but calculating all the values inside the matrix is only of complexity $O(N^2)$. With the generated tree, we calculate the option price.

B.2 A matrix representation

Thus, we will now build the tree as a matrix. The next piece of code is a template structure, it is not yet complete. Please code the missing parts and calculate the stock price.

```
import numpy as np

def buildTree(S, vol, T, N):
  dt = T / N

  matrix = np.zeros((N + 1, N + 1))

  u = 0 # TODO
  d = 0 # TODO

  # Iterate over the lower triangle
```

```
  for i in np.arange(N + 1): # iterate over rows
    for j in np.arange(i + 1): # iterate over columns
      # Hint: express each cell as a combination of up
      # and down moves
      matrix[i, j] = 0 # TODO

  return matrix
```

We can execute the code as follows:

```
sigma = 0.1
S = 80
T = 1.
N = 2

buildTree(S, sigma, T, N)
```

B.3 Calculating the option value

Now that we have the pricing tree, we can use this as input to compute the option value. The next piece of code is a template structure, it is not yet complete. Please code the missing parts and calculate the option price.

```
def valueOptionMatrix(tree, T, r, K, vol):

  dt = T / N

  u = 0 # TODO
  d = 0 # TODO

  p = 0 # TODO

  columns = tree.shape[1]
  rows = tree.shape[0]

  # Walk backward, we start in last row of the matrix

  # Add the payoff function in the last row
  for c in np.arange(columns):
    S = tree[rows - 1, c] # value in the matrix
    tree[rows - 1, c] = 0 # TODO

  # For all other rows, we need to combine from previous rows
  # We walk backwards, from the last row to the first row
  for i in np.arange(rows - 1)[::-1]:
    for j in np.arange(i + 1):
      down = tree[i + 1, j]
      up = tree[i + 1, j + 1]
      tree[i, j] = 0 # TODO
  return tree
```

We can execute the function as follows:

```
sigma = 0.1
S = 80
T = 1.
N = 2

K = 85
r = 0.1

tree = buildTree(S, sigma, T, N)
valueOptionMatrix(tree, T, r, K, sigma)
```

B.4 Plotting

We want to study the correctness of the implementation, thus we want to compare the solution of our algorithm with the analytical answer.
Please create a plot that has on the X-axis the number of steps in the tree (depth), and on the Y-axis the error with respect to the analytical solution.

```
# Play around with different ranges of N and step sizes.
N = np.arange(1,300)

# Calculate the option price for the correct parameters
optionPriceAnalytical = 0 # TODO

# calculate option price for each n in N
for n in N:
    treeN = buildTree(...) # TODO
    priceApproximatedly = valueOption(...) # TODO


# use matplotlib to plot the analytical value
# and the approximated value for each n
```

C The Black-Scholes model

To date, the Black-Scholes model is perhaps the most popular tool for financial derivative valuation. It is a continuous-time model for the evolution of asset prices. Here we will discuss the main assumptions to the model, treat simulation techniques and derive the famous Black-Scholes formula for European options.
Although it is more realistic than the binomial tree, the Black-Scholes model is still a simplified representation of the market. Let again S denote the value of a stock. The main concept here is that the risk-neutral dynamics of the stock price are captured by a *geometric Brownian motion*. This is a stochastic process that is described by the following SDE.

$$dS_t = rS_t dt + \sigma S_t dz_t \tag{10}$$

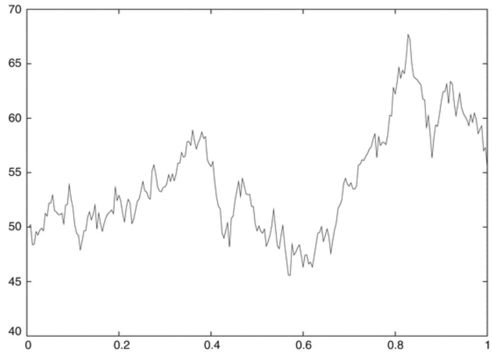


Figure 4: Stock price movements simulated by a geometric Brownian motion. Source: [2].

The drift of the process is constant and equal to the risk-free rate r. The diffusion coëfficient is also constant and referred to as the volatility σ. The process z_t is a *standard Brownian motion* or *Wiener process*. Below we summarize the main assumptions of this model:

1. Cash invested in the money-market yields a continuously compounded interest at a constant rate r.
2. The stock price S_t follows a geometric Brownian motion, with constant drift and volatility.
3. The economy is free of arbitrage.
4. There are no transaction costs for trading shares of stock.

C.1 Derivation Black-Scholes formula

In this section we will derive the analytical formula to price a European call option written on an asset in the Black-Scholes model. This formula is traditionally known as the Black-Scholes formula. Recall that a call option is a contract which gives the holder the right, but not the obligation, to buy an underlying asset S at a specified strike price K at a specified future date T. The pay-off at maturity T of this security is therefore equal to

$$V(T) = (S_T - K)^+ := \max\{S_T - K, 0\}$$

Let $0 \leq t < T$ be any time before maturity. In accordance with the risk-neutral valuation principle, the value of the contract at time t is equal to the expected value of the pay-off, discounted with the risk-free interest rate. This is given by (compare to eq. (5))

$$V(t) = \mathbb{E}\left[e^{-r(T-t)}V(T)\right] = e^{-r(T-t)}\mathbb{E}\left[(S_T - K)^+\right]$$

We will analytically compute $V(t)$ by first deriving the probability density of S_T and secondly evaluating the expectation above.

C.1.1 The distribution of S_T

The main assumption is that the underlying asset is modelled as a geometric Brownian motion. This means that the risk-neutral dynamics of S_t are given by the following stochastic process:

$$dS_t = rS_t dt + \sigma S_t dz_t$$

where r is the constant risk-free rate, σ the constant volatility and z_t a Brownian motion. We can find an exact expression for S_t through an application of Itô's lemma. Itô's lemma says that if a variable x follows a stochastic process of the form

$$dx = a(x,t)dt + b(x,t)dz_t$$

then any smooth function $G(x,t)$ follows the process

$$dG = \left(\frac{\partial G}{\partial x} a(x,t) + \frac{\partial G}{\partial t} + \frac{1}{2}\frac{\partial^2 G}{\partial x^2} b^2(x,t) \right) dt + \frac{\partial G}{\partial x} b(x,t) dz_t$$

Now let $G(S,t) = \log(S)$. Then we have

$$\frac{\partial G}{\partial S} = \frac{1}{S}, \qquad \frac{\partial^2 G}{\partial S^2} = -\frac{1}{S^2}, \qquad \frac{\partial G}{\partial t} = 0$$

and according to Itô's lemma it follows

$$d\log(S_t) = \left(r - \frac{1}{2}\sigma^2 \right) dt + \sigma dz_t$$

Note that on the right hand side, the unknown, stochastic S_t has dropped out of the expression. Now we can simply integrate both sides, which would not have been possible if S_t appeared in the integrand.

$$\begin{aligned} \int_t^T d\log(S_u) &= \int_t^T \left(r - \frac{1}{2}\sigma^2 \right) du + \int_t^T \sigma dz_u \\ \log(S_T) - \log(S_t) &= \left(r - \frac{1}{2}\sigma^2 \right)(T-t) + \sigma(z_T - z_t) \end{aligned}$$

A property of Brownian motion is that any increment is normally distributed as $z_T - z_t \sim \mathcal{N}(0, T-t)$. Hence, if we let Z denote a standard normal random variable, we can write $z_T - z_t \simeq \sqrt{T-t}Z$. We finalise by taking the exponential on both sides of the equation above.

$$\frac{S_T}{S_t} = \exp\left\{ \left(r - \frac{1}{2}\sigma^2 \right)(T-t) + \sigma\sqrt{T-t}Z \right\}$$

It can be concluded that $\frac{S_T}{S_t}$ has a *lognormal distribution.*

C.1.2 Compute the expectation

Now that we have the distribution of S_T, we can proceed by analytically computing the expectation in the expression for $V(t)$. For convenience, define $\tau := T - t$. Also, in the expression for S_T we will write $-Z$ instead of Z, which by its symmetry is equivalent, but will make the computations below easier. Substitution of S_T yields

$$\begin{aligned} V(t) &= e^{-r\tau}\mathbb{E}\left[(S_T - K)^+\right] \\ &= e^{-r\tau}\mathbb{E}\left[\left(S_t \exp\left\{ \left(r - \frac{1}{2}\sigma^2 \right)\tau - \sigma\sqrt{\tau}Z \right\} - K \right)^+ \right] \\ &= \frac{e^{-r\tau}}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left(S_t \exp\left\{ \left(r - \frac{1}{2}\sigma^2 \right)\tau - \sigma\sqrt{\tau}x \right\} - K \right)^+ e^{-\frac{1}{2}x^2} dx \end{aligned}$$

The integrand of the integral above is only non-zero if $S_t \exp\left\{\left(r - \frac{1}{2}\sigma^2\right)\tau - \sigma\sqrt{\tau}x\right\} - K > 0$. This is exactly the case whenever for the integration-variable x we have

$$x < \frac{\log\left(\frac{S_t}{K}\right) + \left(r - \frac{1}{2}\sigma^2\right)\tau}{\sigma\sqrt{\tau}} := d_2$$

It follows that

$$\begin{aligned} V(t) &= \frac{e^{-r\tau}}{\sqrt{2\pi}} \int_{-\infty}^{d_2} \left(S_t \exp\left\{\left(r - \frac{1}{2}\sigma^2\right)\tau - \sigma\sqrt{\tau}x\right\} - K\right) e^{-\frac{1}{2}x^2} dx \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{d_2} e^{-r\tau} S_t \exp\left\{\left(r - \frac{1}{2}\sigma^2\right)\tau - \sigma\sqrt{\tau}x\right\} e^{-\frac{1}{2}x^2} dx - \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{d_2} e^{-r\tau} K e^{-\frac{1}{2}x^2} dx \end{aligned}$$

Let $N(x) := \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{1}{2}u^2} du$ denote the standard normal cumulative distribution function. Then the integral on the right can easily be rewritten as

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{d_2} e^{-r\tau} K e^{-\frac{1}{2}x^2} dx = e^{-r\tau} K N(d_2)$$

The integral on the left is trickier and requires a variable substitution. Define $y = x + \sigma\sqrt{\tau}$, then the integral can be rewritten as

$$\begin{aligned} &\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{d_2} e^{-r\tau} S_t \exp\left\{\left(r - \frac{1}{2}\sigma^2\right)\tau - \sigma\sqrt{\tau}x\right\} e^{-\frac{1}{2}x^2} dx \\ &= \frac{S_t}{\sqrt{2\pi}} \int_{-\infty}^{d_2} \exp\left\{-\frac{1}{2}\left(x + \sigma\sqrt{\tau}\right)^2\right\} dx \\ &= \frac{S_t}{\sqrt{2\pi}} \int_{-\infty}^{d_2 + \sigma\sqrt{\tau}} \exp\left\{-\frac{1}{2}y^2\right\} dy = S_t N(d_1) \end{aligned}$$

where we inherently defined $d_1 := d_2 + \sigma\sqrt{\tau}$. Our final result is what is known as the Black-Scholes formula for a call option:

$$V(t) = S_t N(d_1) - e^{-r\tau} K N(d_2) \tag{11}$$

$$d_1 = \frac{\log\left(\frac{S_t}{K}\right) + \left(r + \frac{1}{2}\sigma^2\right)\tau}{\sigma\sqrt{\tau}} \tag{12}$$

$$d_2 = d_1 - \sigma\sqrt{\tau} \tag{13}$$

C.2 Simulation techniques

In this section we treat simulation techniques for Itô processes. Itô processes form a class of stochastic processes that are commonly used to describe phenomena in physiscs, biology and finance which are subject to randomness. The dynamics of an Itô process X_t are characterized by an SDE of the form

$$dX_t = a(t, X_t)dt + b(t, X_t)dz_t \tag{14}$$

where z_t denotes a standard Brownian motion. A geometric Brownian motion as in eq. (10) is therefore an example of an Itô process, with $a(t, X_t) = rX_t$ and $b(t, X_t) = \sigma X_t$. In integral form, an Itô process can be expressed as

$$X_t = X_0 + \int_0^t a(u, X_u)du + \int_0^t b(u, X_u)dz_u \tag{15}$$

The question is now: how do you simulate trajectories of the process above. In the following sections we provide two approaches.

C.2.1 Exact sampling of an SDE

An explicit solution to the integral of eq. (15) is not always known. If however a solution is available, then we can directly sample trajectories of the Itô process, because its distribution is known. In the specific case of a geometric Brownian motion (GBM), there is an exact solution to the SDE. We derived it in section C.1.1, where we showed that

$$S_T = S_t \exp\left\{ \left(r - \frac{1}{2}\sigma^2 \right)(T-t) + \sigma(z_T - z_t) \right\}$$

The increments $z_T - z_t$ of a standard Brownian motion are independent, Gaussian random variables with distribution $z_T - z_t \sim \mathcal{N}(0, T-t)$. The process can hence be simulated as described in Algorithm 1.

Algorithm 1: Exact simulation GBM

Select $M \in \mathbb{N}$, set $\Delta t = \frac{T}{M}$;
Initialize S_0, the stock price today;
for $m = 1, \ldots, M$ **do**
  Sample $Z_m \sim \mathcal{N}(0, 1)$;
  $S_{m \cdot \Delta t} = S_{(m-1)\cdot\Delta t} \exp\left\{ \left(r - \frac{1}{2}\sigma^2\right)\Delta t + \sigma\sqrt{\Delta t} Z_m \right\}$
end

C.2.2 Discrete Euler method for an SDE

In case an explicit solution to the integral of eq. (15) is not available, one has to settle with an approximation. Perhaps the most intuitive approach for an Itô process is the Euler method. The main idea is that the SDE is discretized. A trajectory of the process can then be approximated on a dicretized time-grid. A discretization of the Itô SDE in eq. (14) is given by

$$\Delta X_t = a(t, X_t)\Delta t + b(t, X_t)\Delta z_t$$

As a property of the Brownian motion, we know that $\Delta z_t := z_{t+\Delta t} - z_t \sim \mathcal{N}(0, \Delta t)$. The accuracy of the Euler discretization depends on the step-size Δt. A smaller Δt implies a higher accuracy, but also requires a larger number of simulation steps. The stock-price process can hence by approximation be simulated as described in Algorithm 2.

Algorithm 2: Approximate simulation GBM with Euler method

Select $M \in \mathbb{N}$, set $\Delta t = \frac{T}{M}$;
Initialize S_0, the stock price today;
for $m = 1, \ldots, M$ **do**
  Sample $Z_m \sim \mathcal{N}(0, 1)$;
  $S_{m\cdot\Delta t} = S_{(m-1)\cdot\Delta t} + rS_{(m-1)\cdot\Delta t}\Delta t + \sigma S_{(m-1)\cdot\Delta t}\sqrt{\Delta t} Z_m$
end

Figure 5 shows a trajectory of a geometric Brownian motion. One line represents the path that is simulated with the exact method. The other two are approximations based on the Euler method, using 100 and 25 steps respectively.

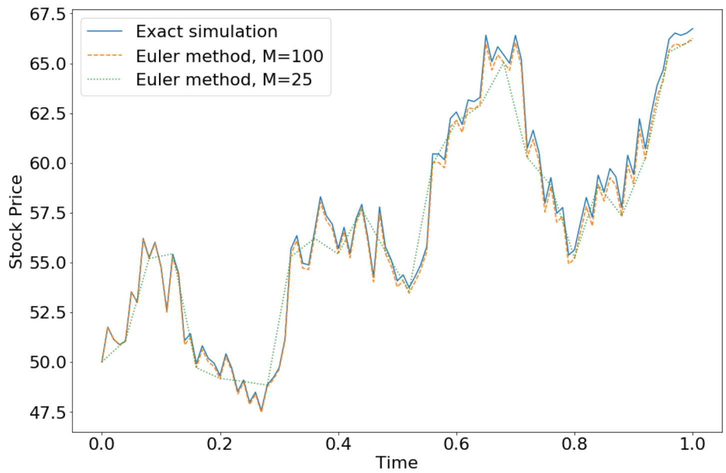


Figure 5: Simulation of a geometric Brownian motion, exact and using the Euler method.

References

[1] S. Shreve, *Stochastic calculus for finance I: the binomial asset pricing model*. Springer Science & Business Media, 2005.

[2] R. Seydel and R. Seydel, *Tools for computational finance*, vol. 3. Springer, 2006.